

UPPAAL MODEL FOR VALIDATION OF THE SOFTWARE DEVELOPMENT PROCESS

ATIRAH SHAHIRA BINTI MAHADI

BACHELOR OF COMPUTER SCIENCE

UNIVERSITI MALAYSIA PAHANG



SUPERVISOR'S DECLARATION

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the degree bachelor of computer science (Software Engineering).

A handwritten signature in black ink, appearing to read "Mezhuyev", is written above a horizontal line.

(Supervisor's Signature)

Full Name : Vitaliy Mezhuyev
Position : Professor
Date : 9 January 2019



STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

A handwritten signature in black ink, appearing to read 'Atirah', written over a horizontal line.

(Student's Signature)

Full Name : Atirah Shahira binti Mahadi

ID Number : CB15006

Date : 9 January 2019

UPPAAL MODEL FOR VALIDATION OF
THE SOFTWARE DEVELOPMENT PROCESS

ATIRAH SHAHIRA BINTI MAHADI

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Software Engineering)

Faculty of Computer Systems and Software Engineering
UNIVERSITI MALAYSIA PAHANG

JANUARY 2019

ACKNOWLEDGEMENTS

I would like to express my greatest thanks and appreciation to my loving parents for providing all the necessities and never ending moral for this research. My siblings, who are very understanding and offers helps even the little things. Also to my lecturer Professor Vitaliy Mezhuyev for willing to spend precious times discussing new ideas, elaborate on the relation of topics for the thesis and utmost supervising that are vital to the process of completing this thesis. Finally, thank you to my friends and classmates who I share all our experiences and time with upon this thesis completion.

ABSTRAK

Pembangunan perisian adalah proses mewujudkan program dan sistem yang berguna untuk meningkatkan kualiti hidup. Terdapat kitaran hidup pembangunan perisian seperti Air Terjun, dan Pembangunan Aplikasi Rapid dan lain-lain. Untuk mengesahkan proses pembangunan perisian, penting untuk menggunakan kaedah formal. Membuktikan sifat sama ada ia beroperasi atau yang berkualiti adalah penting untuk mengelakkan kesilapan dan kemungkinan kemalangan yang boleh meragut nyawa manusia. Satu kajian kes diambil dari sebuah syarikat pembangunan perisian kecil di Malaysia, model yang sama telah dibangunkan dan dianalisis, dan seterusnya disahkan oleh UPPAAL. Kajian ini memberi manfaat kepada syarikat pembangunan perisian untuk merujuk dan membentuk pemahaman tentang pentingnya menggunakan alat rasmi, yang membolehkan membuktikan dan mengesahkan proses perniagaan sebelum pembangunan sistem perisian bermula.

ABSTRACT

Software development is a process of creating useful programs and systems to improve the quality of life. There are Software Development Life Cycles such as Waterfall, and Rapid Application Development among many others. In order to validate the process of software development it is important to use formal methods. Proving properties whether it is operational or quality ones is crucial to avoid errors and possible accidents that can cost human lives. UPPAAL timed automata was selected for development of the model of a software development process in this research. A case study was taken from a small software development company in Malaysia, corresponding model was developed and analysed, and next validated by UPPAAL. This study is beneficial for software development companies to refer and form understanding on the importance of using formal tools, allowing proving and validating the business process before the development of a software system starts.

TABLE OF CONTENT

DECLARATION

TITLE PAGE

ACKNOWLEDGEMENTS **iii**

ABSTRAK **iv**

ABSTRACT **v**

TABLE OF CONTENT **vii**

LIST OF TABLES **ix**

LIST OF FIGURES **x**

LIST OF ABBREVIATIONS **xi**

CHAPTER 1 INTRODUCTION **1**

1.1 INTRODUCTION 1

1.2 PROBLEM STATEMENT 2

1.3 OBJECTIVE 2

1.4 SCOPE 2

1.5 THE IMPORTANCE OF THE PROJECT 3

1.6 THESIS ORGANIZATION 3

CHAPTER 2 LITERATURE REVIEW **4**

2.1 INTRODUCTION 4

2.2 EXISTING MODELS 5

2.2.1 BUSINESS PROCESS MODELLING NOTATION 5

2.2.1.1	BUSINESS PROCESS MODELLING NOTATION SYMBOLS	6
2.2.2	ACTIVITY DIAGRAM 1.x (UML)	9
2.2.2.1	ACTIVITY DIAGRAM 1.x SYMBOLS	10
2.2.3	UPPAAL	13
2.2.3.1	UPPAAL SYMBOLS	15
2.3	COMPARISON OF THREE TYPES OF MODELS	16
CHAPTER 3 METHODOLOGY		17
3.1	INTRODUCTION	17
3.2	METHODOLOGY FRAMEWORK	18
3.3	HARDWARE AND SOFTWARE	189
3.3.1	HARDWARE SPECIFICATIONS	189
3.3.2	SOFTWARE SPECIFICATIONS	20
3.3.3	UPPAAL INTERFACES	20
3.4	GANTT CHART	23
CHAPTER 4 MODEL DEVELOPMENT AND DISCUSSION		24
4.1	INTRODUCTION	24
4.2	MODEL DEVELOPMENT	24
4.3	MODEL DEVELOPMENT AND SIMULATION	28
4.4	CONCLUSION	38
CHAPTER 5 CONCLUSION		1739
5.1	INTRODUCTION	39
5.2	LIMITATIONS	40

5.3	FUTURE WORK	40
	REFERENCES	41
	APPENDICES APPENDIX-A GANTT CHART	42

LIST OF TABLES

Table 2.1	BPMN Symbols	6
Table 2.2	Activity Diagram 1.x Symbols	10
Table 2.3	UPPAAL Symbols	15
Table 2.4	Model Comparisons	16
Table 3.1	Methodology Framework	18
Table 3.2	Hardware Specifications	19
Table 3.3	Software Specifications	20

LIST OF FIGURES

Figure 2.1	A Standard BPMN representations	8
Figure 2.2	Example of Activity Diagram	12
Figure 2.3	UPPAAL Overview	13
Figure 2.4	State and transition	14
Figure 2.5	UPPAAL Simulator Tab	14
Figure 2.6	UPPAAL Lamp Example	15
Figure 3.1	UPPAAL Interface Add Location	20
Figure 3.2	UPPAAL Lamp Interface Add Edge	21
Figure 3.3	UPPAAL Lamp Interface Selection Tool	21
Figure 3.4	UPPAAL Lamp Interface Editing Edge	22
Figure 4.1	Scheme of Software Development Process in a Company	27
Figure 4.2	Team Leader	29
Figure 4.3	System Analysis and Design	30
Figure 4.4	Software Developer Process	31
Figure 4.5	Software Testing	32
Figure 4.6	Software Quality Assurance	34
Figure 4.7	Manager	35
Figure 4.8	Message Sequence	37
Figure A.1	Gantt Chart for PSM 1	42
Figure A.2	Gantt Chart and PSM 2	42

LIST OF ABBREVIATIONS

SDLC	Software Development Life Cycle
UML	Unified Modelling Language
BPMN	Business Process Modelling Notation
CM	Computing Module
GUI	Graphical User Interface
FSM	Finite State Machine

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Developed by Uppsala University, Sweden with collaboration of Aalborg University, Denmark, UPPAAL's initial release was in the year 1995. The program was written in C++ language with Graphical User Interface (GUI) in Java. UPPAAL is operational on multiple operating systems such as Linux, Microsoft Windows and Mac OS X.

UPPAAL is an integrated environment created for the modelling, verification and validation of the processes with the data types and clocks. It is a model checker for real time systems that are modelled as networks in a graphical form. Data types, integers and arrays are important components in UPPAAL.

In software development companies, process modelling is used to control the project development. Since multiple number of methodologies (SDLC) exists and practiced within the organization, usage of formal methods for validation of software development process is needed to help companies. A model based on real cases of software development will be developed using UPPAAL, and the properties and variables will be validated. Example of properties to be tested is meeting deadlines for a project within a range of time.

1.2 PROBLEM STATEMENT

There are many methodologies for software development, but they don't use techniques to model the processes formally. Commonly they are expressed in descriptive from,

which has many drawbacks. The main problem is that software development methodologies use a graphical notation which do not allow validation or proving the process properties.

1.3 OBJECTIVE

The objectives of this research are:

- i. To analyse existing approaches for modelling of software development processes and validation of their properties.
- ii. To develop UPPAAL model of the process of software development in a small company.
- iii. To validate the properties of the developed model.

1.4 SCOPE

The scope of this project is:

- i. The model of the process covers software development process in a small software company in Malaysia.
- ii. Validation of the properties is done by using UPPAAL model checker.
- iii. Users of the proposed model are the team and project managers in software development companies.

1.5 THE IMPORTANCE OF THE PROJECT

The importance of this project is:

Proposed UPPAAL model can be used by project managers in software development companies to optimize software development process and to make sure all of the deliverables are achieved in time.

1.6 THESIS ORGANIZATION

This document consists of three chapters where the first chapter covers the introduction of the project. The topics included in the chapter are introduction, problem statement, objective, scope, and significance of this project.

The next chapter is a literature review that analyzes existing technologies for process modelling and validation available in the current SW industry.

The Chapter 3, there is elaboration on methodology of modelling in this research with justifications. Planning phase also included in this chapter to show the progress of this thesis from beginning to end.

Chapter 4 discusses on the findings based on information and research from previous chapters as well as detailed analysis. Finally, Chapter 5 concludes the research while giving suggestions for future related research. The conclusion, limitation of research, future work sections and reference list finalize the study.

REFERENCES

- Behrmann, G., David, A., & Larsen, K. G. (n.d.). A Tutorial on Uppaal.
- Börger, E. (2012). Approaches to modeling business processes: A critical analysis of BPMN, workflow patterns and YAWL. *Software and Systems Modeling*, 11(3), 305–318. <https://doi.org/10.1007/s10270-011-0214-z>
- Clavel, M., Egea, M., & Torres, V. (n.d.). MOVA : A Tool for Modeling , Measuring and Validating UML Class Diagrams.
- De Giacomo, G., Dumas, M., Maggi, F. M., & Montali, M. (2015). Declarative process modeling in BPMN. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9097, 84–100. https://doi.org/10.1007/978-3-319-19069-3_6
- Grobelna, I., & Grobelny Michał and Adamski, M. (2014). Model Checking of UML Activity Diagrams in Logic Controllers Design. In W. Zamojski, J. Mazurkiewicz, J. Sugier, T. Walkowiak, & J. Kacprzyk (Eds.), *Proceedings of the Ninth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX. June 30 -- July 4, 2014, Brunów, Poland* (pp. 233–242). Cham: Springer International Publishing.
- Johansson, L., Wärja, M., & Kjellin, H. (2014). Graphical modeling techniques and usefulness in the Model Driven Arcitechture : Which are the criteria for a " good " Com Graphical modeling techniques and usefulness in the Model Driven Arcitechture : Which are the criteria for a “ good ” Computer i, (January 2008).
- Uppaal, L. (2009). Uppaal 4.0 : Small Tutorial * 16, (November), 1–11.
- Finite State Machines. Brilliant org. Retrieved 21:58, May 3, 2018, from <https://brilliant.org/wiki/finite-state-machines/>
- Business Process Modelling. BPMN forum. Retrived 15:58, Feb 20, 2018, from <http://bpmnforum.com/bpmn-faq/>
- Activity Diagram. Smartdraw. Retrieved 12:20, Feb 22, 2018 <https://www.smartdraw.com/activity-diagram/>
- Business Process Modelling Notation. Lucidchart. Retrieved 21:02, Feb 22, 2018 <https://www.lucidchart.com/pages/bpmn-symbols-explained>